

## Application Note of NX1 for Using ICE Simulation

**Description:** To illustrate the behavior of NX1 peripheral functions while ICE stops in the process of simulation.

**Reason:** When using NX1\_FDB Ver.A/B or NX1\_ICE Ver.A with NX1\_COB VerA/B to debug and stop in Debug Mode, some of the NX1 peripheral functions won't stop and the values of relevant registers are not those in this pause status.

**Solution:** It is recommended that users transmit the data through the UART interface to the terminal for inspection, instead of using stopping of ICE Debug Mode to inspect registers.

### [Example 1]

Use ADC with Timer Trigger function in the development environment of NX1\_FDB Ver.A/B. When NX1 pauses, the FIFO of ADC register is found to be full and the flag is set. This is because Timer is still working and triggers the convertor of ADC register and result in FIFO is filled up within a short time.

In the above circumstance, if Auto Scan Mode with FIFO is also used, the Channel control will be out of order when NX1 pauses.

However, NX1\_FDB Ver.C and NX1\_ICE Ver.A with NX1\_COB Ver.C, the behavior of peripheral functions in ICE Debug Mode of NX1 is changed to pause while ICE stops, in order to be consistent with the result of Free Run.

Please also notice that using I<sup>2</sup>C Slave Mode in Debug Mode, NX1 is paused but it will still execute the I<sup>2</sup>C control flow. When the Master issues Read commands, only the current first data of FIFO is transmitted and FIFO Count isn't decreased. If Free Run continues to be executed, an error will occur because FIFO Data is shifted.

Please set the breakpoint after I<sup>2</sup>C finishes the transmission. If it has to set the breakpoint in the process of transmission, it is recommend to restart the Debug Mode after checking the status.

The real situation of Free Run: I<sup>2</sup>C FIFO status Data[4] = { 0xF0 ,0xA5 ,0x69 ,0x0F }



ICE Debug Mode: Because EV chip has stopped, and Master needs to access data from NX1 (Slave), only the first data of FIFO is transmitted, as shown below.

